Priebeh celoštátneho kola

Celoštátne kolo 38. ročníka Olympiády v informatike, kategórie A, sa koná v dňoch 22.-25. 3. 2023. Na riešenie úloh druhého, praktického dňa majú súťažiaci 4,5 hodiny čistého času. Akékoľvek pomôcky okrem písacích potrieb (napr. knihy, výpisy programov, kalkulačky) sú zakázané.

Čo má obsahovať riešenie úlohy?

- Skompilovateľný program v podporovanom programovacom jazyku. Ak sa váš program nepodarí na našom testovacom počítači skompilovať a spustiť, bude automaticky hodnotený 0 bodmi.

Hodnotenie riešení druhého (praktického) dňa

Sú tri úlohy. Ku každej úlohe máme pripravených 10 sád testovacích vstupov. Sada vstupov pozostáva z jedného alebo viacerých testovacích vstupov. Za každú sadu vstupov, ktorej všetky vstupy (každý zvlášť) váš program správne vyrieši, získate jeden bod.

Testovanie na každom vstupe prebieha samostatne. Spustíme váš program a na štandardný vstup mu dáme konkrétne vstupné údaje. Hovoríme, že váš program daný vstup vyriešil, ak splní nasledujúce kritériá:

- Skončí skôr ako uplynie stanovený časový limit.
- Neprekročí stanovený pamäťový limit.
- Skončí korektne, nie chybou počas behu.
- Dáta, ktoré vypíše na štandardný výstup, tvoria korektný výstup, zodpovedajúci danému vstupu.
- Nebude používať žiadne funkcie zakázané kvôli bezpečnosti testovacieho systému.

Počas súťaže môžete priebežne odovzdávať svoje riešenia. Odovzdané riešenie bude otestované a dozviete sa svoj bodový zisk. (V prípade preťaženia testovača môžu organizátori obmedziť toto priebežné testovanie na vhodnú podmnožinu všetkých testovacích dát.)

Po ukončení súťaže zoberieme pre každú úlohu váš posledný odovzdaný program a ten otestujeme na všetkých testovacích vstupoch. Vaše výsledné body za úlohu budú body získané týmto programom.

Sady vstupov sú navrhované tak, aby každé korektné riešenie získalo nejaké body, bez ohľadu na to, ako pomalé je. Bližšie informácie o testovacích dátach nájdete na konci zadania každej úlohy.

A-III-4 Diaľnica v Slovakistane

Andrejovi sa podarilo získať prácu v slovakistanskej Náhodnej spoločnej diaľničnosti (*kašeľ* cez sestru *kašeľ*). Hneď v prvý deň dostal za úlohu dohliadnuť na dokončenie diaľnice D1, čo je neľahký údel – veľa vybavovačiek, papierovačiek a iných -ovačiek; proste veľa dokumentov, lietajúcich zľava doprava a sprava doľava.

Z úradov prichádzajú oznamy o stavoch jednotlivých častí diaľnice. Oznamy sú jednoduché, každý obsahuje informáciu o tom, ktoré nové časti sú vo výstavbe. Tu ale prichádza Andrejov problém. Keďže je v úradných dokumentoch taký neporiadok, často sa stáva, že tá istá časť diaľnice je stavaná viacerými firmami naraz.

Samozrejme, žiadna časť diaľnice nikdy nebude dokončená – akonáhle raz začne byť rozostavaná, bude už rozostavaná na veky vekov.

Andrejova úloha je, slovami jeho nadriadených, jednoduchá. Stačí po každom dokumente, ktorý mu pristane na stole, vedieť povedať, koľko súvislých úsekov diaľnice je aktuálne v rozostavanom stave.

Súťažná úloha

Andrej si diaľnicu D1 predstavuje ako úsečku rozdelenú na $n + 1$ *častí* očíslovaných od 0 po n. Spracovať má postupne q oznamov. Každý oznam vieme popísať číslami a_i a b_i, ktoré určujú *súvislý interval* častí, ktoré sú odteraz vo výstavbe. Konkrétne, sú to všetky časti od a_i po b_i vrátane.

Po každom ozname musí Andrej svojim nadriadeným povedať, koľko súvislých *úsekov* diaľnice je vo výstavbe. Úsek je súvislý interval niekoľkých po sebe idúcich rozostavených častí diaľnice, pričom na každom konci úseku je buď nerozostavaná časť diaľnice alebo začiatok/koniec celej diaľnice.

Formát vstupu a výstupu

Na prvom riadku vstupu sa nachádzajú dve celé čísla n a q – číslo poslednej časti diaľnice a počet oznamov, ktoré postupne pristáli na Andrejovom stole.

Nasleduje q riadkov, každý z nich obsahuje jeden oznam: dve celé čísla a_i a b_i hovoriace, že všetky časti s číslom x, kde $a_i \leq x \leq b_i$, sú odteraz rozostavané.

Na výstup vypíšte q riadkov. Na i-ty riadok vypíšte počet súvislých rozostavaných úsekov diaľnice po spracovaní i-teho oznamu.

Bodovanie

Vo všetkých vstupoch platí $1 \leq n \leq 10^9$ a $1 \leq q \leq 2 \cdot 10^5$. Vo všetkých vstupoch pre všetky i platí $0 \leq a_i \leq b_i \leq n$. Sú štyri sady vstupov:

- V prvej (2 body) navyše platí $n, q \leq 1000$.
- V druhej (3 body) pre všetky i platí $b_i - a_i \leq 10$.
- V tretej (2 body) platí $n \leq 2 \cdot 10^5$.
- Vo štvrtej (3 body) nie sú žiadne ďalšie obmedzenia.

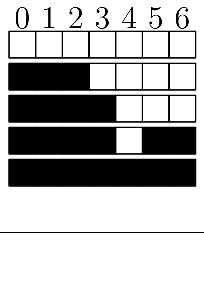
Príklad

vstup

```
6 4
0 2
1 3
5 6
4 4
```

output

```
1
1
2
1
```



Prvý a druhý oznam rozostavajú prelínajúce sa intervaly. Jediný rozostavaný úsek po druhom ozname je ten na intervale 0 až 3. Tretí oznam pridá ďalší úsek od 5 po 6. Posledný oznam následne spojí dva existujúce úseky čím vznikne jediný rozostavaný úsek od 0 po 6.

A-III-5 Planéta B-647

Ôsma planéta bola veľmi neobyčajná. Mala $n = 10^5$ vrcholov a každý vrchol bol spojený s práve dvoma inými vrcholmi, tvoriac nádherný cyklus. Býval na nej fyzik, ktorý zrovna zúrivo mačkal tlačítko na nejakej podivnej aparatúre.

„Čo robíš?“ opýtal sa ho malý princ.

„Nedávno som čítal knihu *Using the Borsuk-Ulam Theorem*, kde písali, že na obyčajných planétach sa vždy dajú nájsť dve protiľahlé miesta, na ktorých je rovnaká teplota.“ odpovedal fyzik. „Snažím sa zistiť, či to platí aj na mojej neobyčajnej planéte.“

„A čo musíš urobiť, aby si to zistil?“ odvetil malý princ, nerozumejúc.

„Musím odmerať teplotu v každom vrchole,“ povedal fyzik, „ale tento teplomer meria veľmi pomaly, a už ma nebaví toľko čakať.“

„Pomôžem ti,“ povedal malý princ. Naznačujúc fyzikovi, nech si sadne do tieňa, sa chopil prístroja. „Aspoň je to väčšia zábava, než u toho obchodníka,“ pomyslel si malý princ.

Nanešťastie, aj malému princovi sa meranie asi po piatich minútach dosť zunovalo. „Keby som len vedel postupovať nejako šikovnejšie, aby som nemusel merať teplotu v každom vrchole.“

Súťažná úloha

Vrcholy planéty sú očíslované od 0 po $n-1$ proti smeru hodinových ručičiek. Číslo n **je rovné 100 000**.

Pre každé i je vo vrchole i teplota t_i, pričom platí, že teplota v susedných vrcholoch sa vždy líši práve o 1 stupeň. (Formálne, pre $0 \leq i < n-1$ platí $|t_i - t_{i+1}| = 1$ a navyše tiež $|t_{n-1} - t_0| = 1$.)

Pomôžte malému princovi na čo najmenej meraní nájsť $0 \leq i < n/2$ také, že $t_i = t_{i+n/2}$, alebo zistiť, že také i neexistuje.

Interakcia

Vašou úlohou je napísať program, ktorý bude komunikovať s „malým princom“ (naším programom). Váš program bude striedavo písať údaje na štandardný výstup a čítať údaje zo štandardného vstupu.

V každej *iterácii* položíte jednu otázku tak, že vypíšete riadok tvaru „0 x“, kde $0 \leq x < n$ je číslo vrcholu, ktorého teplotu má malý princ odmerať. Následne prečítate odpoveď malého princa, jedno celé číslo t_x. Vždy bude platiť, že $-273 \leq t_x \leq 10^9$.

Pozor na to, že po každej otázke je treba vyprázdniť buffer štandardného výstupu, aby sa vypísané dáta ihneď dostali ku malému princovi:

- v C a C++ pri používaní `stdio` pomocou `fflush(stdout);`
- v C++ pri používaní `iostream` pomocou `cout « flush;`
- v Pascale pomocou `flush(output);`
- V Pythone stačí funkcii `print` pridať argument `flush=True`

Až si budete istý odpoveďou, vypíšte jeden riadok v tvare 1 x, kde buď $0 \leq x < n/2$ je číslo takého vrcholu, že $t_x = t_{x+n/2}$, alebo $x = -1$, ak žiadny taký vrchol neexistuje. Následne úspešne ukončite svoj program.

Ukážkový testovač

Na testovači bude k dispozícií ukážková implementácia „malého princa“, ktorú môžete použiť pri testovaní svojho programu. Tento program najskôr prečíta zo súboru `planeta.in` popis teplôt vrcholov planéty a potom vyššie uvedeným spôsobom komunikuje pomocou svojho štandardného vstupu a výstupu.

Súbor `planeta.in` má obsahovať dva riadky. Na prvom je prirodzené číslo n popisujúce počet vrcholov planéty. Číslo n musí byť párne (inak by neexistovali proti sebe stojace vrcholy) a musí platiť $2 \leq n \leq 10^5$. Na druhom riadku je n medzerou oddelených celých čísiel medzi -273 a 10^9: teploty t_0 až t_{n-1}.

Príklad

V tomto príklade je $n = 10^5$, všetky vrcholy s párnym číslom majú teplotu 0 a všetky vrcholy s nepárnym číslom teplotu 1. Následuje príklad komunikácie malého princa s Vaším programom. Číta sa zhora dolu:

váš program	*malý princ*	*komentár*
`0 0`		Aká je teplota vo vrchole 0?
	`0`	$t_0 = 0$
`0 1`		Aká je teplota vo vrchole 1?
	`1`	$t_1 = 1$
`0 50000`		Aká je teplota vo vrchole 50 000?
	`0`	$t_{50\,000} = 0$
`1 0`		Už viem riešenie, vrcholy 0 a $0 + n/2 = 50\,000$ majú rovnakú teplotu.

Bodovanie

Program, s ktorým budete interagovať („malý princ“) sa nesnaží byť zákerný. V každom teste sú teploty vo všetkých vrcholoch pevne zvolené a malý princ Vám tieto teploty postupne prezrádza. Môžete predpokladať, že na každú otázku vie odpovedať v rozumne malom konštantnom čase.

Vo všetkých testovacích vstupoch má planéta **presne $\boldsymbol{n = 10^5}$** vrcholov a všetky teploty sú celé čísla medzi -273 a 10^9. Váš program bude spustený na viacerých testovacích vstupoch. Ak na niektorom z nich odpovie nesprávne, získa 0 bodov. Inak označme ako d najväčší počet otázok vášho programu naprieč všetkými testovacími vstupmi. Počet bodov, ktoré dostanete, bude závisieť od hodnoty d. Tabuľka nižšie popisuje, koľko bodov dostanete, pokiaľ hodnota d nepresiahne danú hranicu:

body	d
1	100 000
2	50 000
3	26 000
4	5 000
5	2 000
6	1 000
7	350
8	100
9	34
10	32

A-III-6 Marshmallows

Vo výklade obchodíku pána Jonatána sa v nedeľu zjavil krásny strom. Má n vrcholov a $n - 1$ hrán. Hrany stromu sú špajle a v každom vrchole je buď marshmallow (mňam!) alebo plastelína.

Timka zbožňuje marshmallowy, rozhodla sa preto, že len čo v pondelok ráno pán Jonatán otvorí, príde a tento strom si kúpi. Má to ale pár drobných problémov.

Prvým problémom je, že sa jej nechce zbytočne nosiť veľa plastelíny. Pred kúpou stromu teda nejaké časti z neho odoberie a zahodí.

Druhým problémom ale je, že Timka v pondelok nesie do školy aj svoj projekt z výtvarnej a bude mať voľnú len jednu ruku, v ktorej vie bezpečne odniesť len jeden predmet. Po tom, ako nejaké časti stromu zahodí, jej teda musí ostať jeden súvislý podstrom.

No a posledným problémom je, že tesne pred tým, ako pán Jonatán otvorí, nalepí na jeden z vrcholov cenovku. Tú Timka musí na strome nechať, aby jej pri pokladni mohol pán Jonatán strom nablokovať.

Súťažná úloha

Pomôžte Timke pripraviť sa na nákup.

Podstrom daného stromu je jeho ľubovoľný súvislý podgraf – teda pre ľubovoľné $k > 0$ ľubovoľná sada k vrcholov takých, že medzi nimi vedie $k - 1$ špajlí, ktoré ich držia pokope.

Kvalita podstromu je rovná rozdielu počtu marshmallowov a počtu plastelín v ňom.

Pre každú možnú polohu cenovky zistite najväčšiu možnú kvalitu podstromu, ktorý si Timka môže kúpiť.

Formát vstupu a výstupu

V prvom riadku vstupu je celé číslo n udávajúce počet vrcholov. Tie číslujeme od 0 po $n - 1$.

V druhom riadku je postupnosť n znakov $t_0 t_1 \dots t_{n-1}$. Znak t_i je `M` ak je vo vrchole i marshmallow, resp. `P` ak je tam plastelína.

Zvyšok vstupu tvorí $n - 1$ riadkov. V každom z nich sú čísla dvoch vrcholov, ktoré sú spojené špajľou. Je zaručené, že tieto riadky popisujú strom.

Postupne pre každé i od 0 po $n-1$ vypíšte na samostatný riadok najväčšiu možnú kvalitu podstromu, ktorý obsahuje vrchol i.

Obmedzenia a hodnotenie

Vo všetkých vstupoch platí $2 \leq n \leq 10^6$. Je päť sád vstupov. Platia v nich rôzne dodatočné obmedzenia.

Hodnota d označuje maximálny stupeň vrcholu v našom strome.

V prvej sade (1 bod) platí $n \leq 10$.

V druhej sade (3 body) platí $n \leq 5\,000$.

V tretej sade (2 body) platí $d \leq 2$.

V štvrtej sade (2 body) platí $d \leq 50$.

V piatej sade (2 body) nie sú žiadne dodatočné obmedzenia.

Príklady

vstup

```
6
MMMPPM
0 1
0 2
0 3
3 4
4 5
```

výstup

```
3
3
3
2
2
2
```

vstup

```
5
MPPPP
0 1
1 2
2 3
3 4
```

výstup

```
1
0
-1
-1
-1
```

TRIDSIATY ÔSMY ROČNÍK OLYMPIÁDY V INFORMATIKE

Príprava úloh: Michal Anderle, Michal Forišek, Jakub Šimo
Recenzia: Michal Forišek
Slovenská komisia Olympiády v informatike
Vydal: IUVENTA – Slovenský inštitút mládeže, Bratislava 2023